Hayashi-Chapter 4

```
#pmayav2
# clear memory
rm(list=ls())
#import data
greene <- read.csv("~/Desktop/Econometrics/4_greene.csv")
#View(greene)
col_names <- colnames(greene)
for( i in 1:length(greene))
  assign(col_names[i],greene[[i]])

nobs <- length(greene[[1]])
FUEL_SHARE <- 1-LABOR-CAPITAL
```

#QUESTION B

```
#CONSTRAINED SYSTEM
#vector of regressors (zi)
Z <- cbind(rep(1,nobs), log( PL/PK ), log( PF/PK), log(Q))
X <- Z

#equation-by-equation OLS
#and use the residuals to calculate sigma_hat
source('~/Desktop/Econometrics/user-defined functions/plain_vanilla_OLS_f.R')
s1<- plain_vanilla_OLS_f(LABOR, X, flag_print = 0) #labor share
resid_s1 <- s1$e
s2<- plain_vanilla_OLS_f(CAPITAL, X, flag_print = 0) #capital share
resid_s2 <- s2$e
s3<- plain_vanilla_OLS_f(FUEL_SHARE, X, flag_print = 0) #fuel share
resid_s3 <- s3$e
list_residuals <- list(resid_s1, resid_s2, resid_s3)

#Sigma hat from equation-by-equation OLS
#(An estimate of the 3x3 error covariance matrix Z)
Sigma_hat <- matrix(nrow=3, ncol=3)
for(i in 1:3)
  for ( j in 1:3)
    Sigma_hat[i,j] <- sum(list_residuals[[i]]*list_residuals[[j]])/nobs
Sigma_hat

##               [,1]          [,2]         [,3]
## [1,]  0.0017266283 -0.0001711457 -0.001555483
## [2,] -0.0001711457  0.0025280399 -0.002356894
## [3,] -0.0015554825 -0.0023568942  0.003912377

#extract the appropriate submatrix from Sigma_hat = Sigma_hat_star (2x2
matrix)
```

```
Sigma_hat_star <- Sigma_hat[c(1,3),c(1,3)]
Sigma_inv <- solve(Sigma_hat_star)    #2x2 matrix
Sigma_inv

##          [,1]     [,2]
## [1,] 902.3645 358.7620
## [2,] 358.7620 398.2357

s_1 <- 0
s_2 <- 0
s_xx <- 0
s_xy <- 0
s_xz <- 0
Zi <- matrix( nrow = 2, ncol = 7 )
for (i in 1:nobs){
  Zi[1,] <- c(1, 0, log(PL/PK)[i], log(PF/PK)[i], 0 , log(Q)[i], 0 )
  Zi[2,] <- c(0, 1, 0 , log(PL/PK)[i], log(PF/PK)[i],0, log(Q)[i])
  yi <- rbind(LABOR[i],(1-LABOR-CAPITAL)[i])
  Xi <- c(1, log(PL/PK)[i], log(PF/PK)[i], log(Q)[i])
  s_1 <- s_1 + (t(Zi) %*% Sigma_inv %*% Zi)/nobs  #4.6.18
  s_2 <- s_2 + (t(Zi) %*% Sigma_inv %*% yi)/nobs  #4.6.17
  s_xx <- s_xx + (Xi %*% t(Xi))/nobs
  s_xy <- s_xy + (yi %x% Xi)/nobs
  s_xz <- s_xz + (Zi %x% Xi)/nobs                   #4.6.16
}

avar_hat <- solve(s_1)      #4.6.9
#Random-effects estimate of the seven free parameters
delta_hat_re <- avar_hat %*% ( s_2 )      #4.6.8

rownames(delta_hat_re) <- c("alpha_1", "alpha_3", "gamma_11", "gamma_13",
"gamma_33", "gamma_1Q", "gamma_3Q")
delta_hat_re

##                 [,1]
## alpha_1  -0.13151119
## alpha_3   0.81337544
## gamma_11  0.08362500
## gamma_13 -0.06041580
## gamma_33  0.15938528
## gamma_1Q -0.02115260
## gamma_3Q  0.02973863

#standard errors
SE<- (sqrt(diag(avar_hat/nobs)))
SE<- as.data.frame(SE)
rownames(SE) <- c("alpha_1", "alpha_3", "gamma_11", "gamma_13", "gamma_33",
"gamma_1Q", "gamma_3Q")
SE
```

```
##                    SE
## alpha_1  0.105605969
## alpha_3  0.093557988
## gamma_11 0.019975813
## gamma_13 0.015411984
## gamma_33 0.023113456
## gamma_1Q 0.002474827
## gamma_3Q 0.003724804

#RANDOM EFFECT ESTIMATES table
#creating table
r_effects <- matrix(0, nrow=12, ncol=3)
colnames(r_effects) <- c('Point est', 'SE', 't-value')
rownames(r_effects) <- c("alpha_1", "alpha_2", "alpha_3", "gamma_11",
"gamma_12", "gamma_13", "gamma_22", "gamma_23", "gamma_33", "gamma_1Q",
"gamma_2Q", "gamma_3Q")

#calculating point est column
r_effects[c(1,3,4,6,9,10,12),1] <- delta_hat_re
r_effects[2,1] <- 1 - r_effects[1,1] - r_effects[3,1] #alpha_2
r_effects[5,1] <- - r_effects[4,1] - r_effects[6,1] #gamma_12
r_effects[8,1] <- - r_effects[6,1] - r_effects[9,1] #gamma_23
r_effects[7,1] <- - r_effects[5,1] - r_effects[8,1] #gamma_22
r_effects[11,1] <- - r_effects[10,1] - r_effects[12,1]; #gamma_2Q

#calculating SE column
r_effects[c(1,3,4,6,9,10,12),2] <- sqrt(diag(avar_hat/nobs))
r_effects[2,2] <- sqrt((t(c(1,1)) %*% avar_hat[c(1,2),c(1,2)] %*%
c(1,1))/nobs)
r_effects[5,2] <- sqrt((t(c(1,1)) %*% avar_hat[c(3,4),c(3,4)] %*%
c(1,1))/nobs)
r_effects[7,2] <- sqrt((t(c(1,2,1)) %*% avar_hat[c(3,4,5),c(3,4,5)] %*%
c(1,2,1))/nobs)
r_effects[8,2] <- sqrt((t(c(1,1)) %*% avar_hat[c(4,5),c(4,5)] %*%
c(1,1))/nobs)
r_effects[11,2] <- sqrt((t(c(1,1)) %*% avar_hat[c(6,7),c(6,7)] %*%
c(1,1))/nobs)

#calculating t-value column
r_effects[,3] <- r_effects[,1] /r_effects[,2]

print(r_effects)

##             Point est          SE   t-value
## alpha_1  -0.131511190 0.105605969 -1.245301
## alpha_2   0.318135746 0.084897658  3.747285
## alpha_3   0.813375444 0.093557988  8.693811
## gamma_11  0.083624998 0.019975813  4.186313
## gamma_12 -0.023209197 0.015918973 -1.457958
## gamma_13 -0.060415801 0.015411984 -3.920053
```

```
## gamma_22  0.122178679 0.019744775  6.187899
## gamma_23 -0.098969482 0.017221641 -5.746809
## gamma_33  0.159385284 0.023113456  6.895779
## gamma_1Q -0.021152598 0.002474827 -8.547101
## gamma_2Q -0.008586037 0.002994114 -2.867638
## gamma_3Q  0.029738634 0.003724804  7.983947
```

```
source("~/Desktop/Econometrics/multiple_gmm.R")
library(magic)
```

```
library(matlib)
```

```
M<-2
Z <- cbind(rep(1,nobs), log( PL/PK ), log( PF/PK), log(Q))
y <- cbind(LABOR, FUEL_SHARE)
X <- cbind(1, log(PL/PK), log(PF/PK), log(Q))

##compute S_hat (W)
e_1 <- c(resid_s1) # convert to vector
e_3 <- c(resid_s3) # convert to vector
temp_1 <- e_1*X
temp_3 <- e_3*X
S_1_1 <- t(temp_1)%*%temp_1      #4x4 matrix
S_1_3 <- t(temp_1)%*%temp_3      #4x4 matrix
S_3_1 <- t(temp_3)%*%temp_1      #4x4 matrix
S_3_3 <- t(temp_3)%*%temp_3      #4x4 matrix
s_hat1 <-cbind(S_1_1,S_1_3)
s_hat2 <-cbind(S_3_1,S_3_3)

S_hat<- rbind(s_hat1,s_hat2)     #8x8 matrix
W <-S_hat

multiple_gmm_f(y, X, Z, W, M, flag_print=1)
```

```
##             [,1]
## [1,] 0.766795372
## [2,] 0.006583990
## [3,] 0.105627180
## [4,] 0.008575455
## [5,] 0.765773902
## [6,] 0.006620699
## [7,] 0.105480961
## [8,] 0.008563439
```

```
## ***** Multiple GMM *****
##  Number of Observations: 99
##  Degree of Freedom (Km-Lm): 0
##  Degree of Freedom (Z): 95
##  Centered R-squared 1: -118.7757 -5.244137
##  Centered R-squared 2: 1.080543 -2.011547
```

```
##  Standard Error of the Equation 1: 0.6514755
##  Standard Error of the Equation 1: 0.1626118
##  Sum of Squared Residuals 1: 40.31993
##  Sum of Squared Residuals 2: 2.512046
##  Significance level P 1: 4.986664e-35 0.5352191 0.5611777 0.9101461
##  Significance level P 2: 0 0.01248936 0.02008133 0.6516312
##  Sargan's Statistic 1: 5.647413e-07
##  Sargan's Statistic 2: 8.821656e-09
```

```
#USING BUILT IN FUNCTIONS
#3 regressions calculated with pooled OLS
library(systemfit)

library(dplyr)

library(skimr)


dataframe <- greene

eq1 <- LABOR ~ log(PL / PF) + log(PK / PF) + log(Q)
eq2 <- CAPITAL ~ log(PL / PF) + log(PK / PF) + log(Q)
eq3 <- FUEL_SHARE ~ log(PL / PF) + log(PK / PF) + log(Q)
p_ols <- systemfit(list(LABOR = eq1, CAPITAL = eq2, FUEL = eq3),
                         method = "OLS",
                         data= dataframe)
summary(p_ols)
```

```
##
## systemfit results
## method: OLS
##
##           N  DF      SSR detRCov   OLS-R2 McElroy-R2
## system 297 285 0.808537       0 0.478603       -Inf
##
##           N DF      SSR      MSE     RMSE       R2   Adj R2
## LABOR   99 95 0.170936 0.001799 0.042418 0.492211 0.476176
## CAPITAL 99 95 0.250276 0.002634 0.051327 0.341288 0.320486
## FUEL    99 95 0.387325 0.004077 0.063852 0.535658 0.520995
##
## The covariance matrix of the residuals
##                 LABOR      CAPITAL        FUEL
## LABOR    0.001799328 -0.000178352 -0.00162098
## CAPITAL -0.000178352  0.002634484 -0.00245613
## FUEL    -0.001620977 -0.002456132  0.00407711
##
## The correlations of the residuals
##               LABOR    CAPITAL      FUEL
## LABOR    1.0000000 -0.0819171 -0.598474
## CAPITAL -0.0819171  1.0000000 -0.749424
## FUEL    -0.5984741 -0.7494243  1.000000
```

```
##
##
## OLS estimates for 'LABOR' (equation 1)
## Model Formula: LABOR ~ log(PL/PF) + log(PK/PF) + log(Q)
##
##                Estimate  Std. Error  t value   Pr(>|t|)
## (Intercept) -0.15886387  0.11337397 -1.40124    0.16440
## log(PL/PF)   0.08989280  0.02192023  4.10091 8.6808e-05 ***
## log(PK/PF)  -0.03190131  0.01970889 -1.61863    0.10884
## log(Q)      -0.02114450  0.00252641 -8.36937 4.9871e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.042418 on 95 degrees of freedom
## Number of observations: 99 Degrees of Freedom: 95
## SSR: 0.170936 MSE: 0.001799 Root MSE: 0.042418
## Multiple R-Squared: 0.492211 Adjusted R-Squared: 0.476176
##
##
## OLS estimates for 'CAPITAL' (equation 2)
## Model Formula: CAPITAL ~ log(PL/PF) + log(PK/PF) + log(Q)
##
##                Estimate  Std. Error  t value   Pr(>|t|)
## (Intercept)  0.23524714  0.13718477  1.71482  0.0896387 .
## log(PL/PF)  -0.00686973  0.02652391 -0.25900  0.7961946
## log(PK/PF)   0.11224367  0.02384815  4.70660 8.5582e-06 ***
## log(Q)      -0.00854243  0.00305701 -2.79437  0.0062908 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.051327 on 95 degrees of freedom
## Number of observations: 99 Degrees of Freedom: 95
## SSR: 0.250276 MSE: 0.002634 Root MSE: 0.051327
## Multiple R-Squared: 0.341288 Adjusted R-Squared: 0.320486
##
##
## OLS estimates for 'FUEL' (equation 3)
## Model Formula: FUEL_SHARE ~ log(PL/PF) + log(PK/PF) + log(Q)
##
##                Estimate  Std. Error  t value   Pr(>|t|)
## (Intercept)  0.92361673  0.17066100  5.41200 4.6613e-07 ***
## log(PL/PF)  -0.08302307  0.03299636 -2.51613  0.0135421 *
## log(PK/PF)  -0.08034237  0.02966764 -2.70808  0.0080265 **
## log(Q)       0.02968692  0.00380299  7.80620 7.6923e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.063852 on 95 degrees of freedom
## Number of observations: 99 Degrees of Freedom: 95
```

```
## SSR: 0.387325 MSE: 0.004077 Root MSE: 0.063852
## Multiple R-Squared: 0.535658 Adjusted R-Squared: 0.520995
```

#QUESTION D

```
#Sargan's statistic for the constrained system
gn_re <- (s_xy) - (s_xz) %*% delta_hat_re
s_hatt<- Sigma_hat_star %x% s_xx
#sargans from proposition 4.7c
sargans <- nobs * ( t(gn_re) %*% solve(s_hatt) %*% gn_re )
sargans

##          [,1]
## [1,] 0.633131

#p-value
1-pchisq(sargans, 2*4-7)    #(MK-L)

##           [,1]
## [1,] 0.4262092

#Sargan's statistic for the unconstrained system
Sigma_hat_star1 <- Sigma_hat[c(1,2),c(1,2)]

#sargans from proposition 4.7c
gn_re1 <- (s_xy) - (s_xz) %*% delta_hat_re
s_hatt1<- Sigma_hat_star1 %x% s_xx
sargans1 <- nobs * ( t(gn_re1) %*% solve(s_hatt1) %*% gn_re1 )
sargans1

##          [,1]
## [1,] 0.977571

#p-value
1-pchisq(sargans1, 2*4-7)

##           [,1]
## [1,] 0.3227992
```

#QUESTION E

```
#Wald test of symmetry in the unconstrained system (s1 & s3)
r <- c(0,0,1,0,0,-1,0,0)
coeff <- c(s1$b,s3$b)   #coefficients of s1 and s3
zz<- t(Z) %*% Z/nobs
avar_nr <- Sigma_hat_star %x% solve(zz)

Wald_statistic <- nobs*(coeff[3]-
coeff[6])%*%solve(r%*%avar_nr%*%r)%*%(coeff[3]-coeff[6])
```

```r
#equal to Sargan's statistic for the constrained system  (0.63313)
Wald_statistic

##          [,1]
## [1,] 0.633131
```

#QUESTION F

```r
#Average labor-capital substitution elasticity over the 99 firms.
labor <- Z %*% delta_hat_re[c(1,3,4,6)]
fuel <- Z %*% delta_hat_re[c(2,4,5,7)]
capital <- 1 - labor - fuel

sub_elasticity <- mean(r_effects[5,1]/(labor*capital) + 1)
sub_elasticity

## [1] 0.1695123
```